

REMARKS

Claims 1-15 and 17-32 are pending. In the previous office action, claims 1-15 and 17-32 including independent claims 1, 14, 15, and 17 were rejected under 35 U.S.C. 102(b) as being unpatentable over Albrecht. The Examiner indicated that in response to the Applicants' arguments, a new grounds of rejection is being set forth. It is believed that the Examiner is now attempting to use different portions of Albrecht (USP 5,835,594) to reject the independent claims.

It should be noted that some recitation the Examiner is using Albrecht to anticipate are not in the claims. For example, the Examiner argues that Albrecht (col. 3, lines 6-15) describes "enabling the user logic if the configurable device authorization and the secure device authorization code are identical." This recitation is not extant in any pending claims. Claim 1 does recite "enabling the user logic if the configurable device authorization code corresponds to the secure device authorization code." However, the authorization codes do not have to be identical.

The independent claims and Albrecht are directed at fundamentality different ideas. Albrecht attempts to prevent unauthorized write access to a protected flash memory. Albrecht suggests that write data should have an electronic signature. One digest is generated using the data itself and another digest is generated using the signature itself. If the digests match, authorization is given to write the data to the FLASH memory. (Figure 2) By contrast, the independent claims relate to allowing authorized use of Intellectual Property cores on programmable chips. According to various embodiments, these cores are licensed pieces of intellectual property that allow a designer to more easily implement components such as Fast Fourier Transform cores or Digital Signal Processing cores on a programmable chip. These cores are implemented using resources such as Look Up Tables (LUTs) on a programmable chip.

"To make an illegal copy of the circuit design, as implemented in a configurable logic device, one need only make a copy of the bitwise representation stored in the storage device. This can be done by copying the bitstreams transmitted externally between a configurable device and the device installing the configuration data and using the copies bitstream with a copied configurable device. Thus, the copied bitwise representation can be illegally used with other

programmable logic devices. Therefore, it is desirable to make it more difficult to copy the bitwise representation of the circuit design.” (page 3, lines 23-30)

Independent claim 17 recites user logic that is disabled upon implementation on the configurable device. Albrecht does not have any user logic that is disabled upon implementation on the configurable device. The Examiner may argue that the write data written to FLASH memory is implemented on the configurable device. However, Albrecht describes checking for authorization before write access is granted. Consequently, there is no user logic disabled upon implementation on the configurable device.

Independent claims 1, 13, and 14 have been amended to recite disabling user logic provided for implementation of the configuration data after it is loaded onto the configurable device. Albrecht does not teach or suggest disabling user logic for implementation of the configuration data after it is loaded onto the configurable device. The Applicants respectfully request that the Examiner identify with specificity what is user logic, what is configuration data, and when user logic is disabled after it is loaded on onto the configurable device. It is somewhat confusing to Applicants because the Examiner appears to say that BIOS updates are configuration data but also says the BIOS is implemented on user logic in the form of a computer processor. It is not clear to Applicants whether the Examiner saying that the user logic is a computer processor, software code for implementation on a computer processor, or BIOS code for implementation on a computer processor.

It is respectfully submitted that none of these are user logic. If the BIOS update is user logic, the BIOS update is at no point disabled after implementing the BIOS update on the BIOS in Albrecht. If the BIOS code for implementation on a computer processor, the BIOS code is not at any time disabled after implementing the BIOS code on the computer processor or on the BIOS. If the software code is user logic, the software code is at not point disabled after impelmetning the software code on the BIOS or on the processor. Furthermore, the flash security circuit is also not user logic and is not at any point disabled. “Flash security circuit 226 protects FLASH memory 224 from unauthorized write accesses, by keeping FLASH memory 224 write disabled, and generating an SMI to invoke the secured system BIOS write data authentication functions in system management memory 222 to authenticate the write data, whenever it enables FLASH memory 224 for a write access.” (column 4, lines 25-30) It is contemplated that certain lines of flash memory could be locked by writing lock bits in various

flash memory lines. Consequently, certain lines of memory may be disabled, but no flash security circuitry is disabled. The Applicants respectfully request that the Examiner specify what is user logic in Albrecht, what is configuration data in Albrecht, what the configurable device is, and when the user logic in Albrecht is disabled after it is implemented on the configurable device.

Furthermore, dependent claims recite a programmable logic device. Albrecht does not teach any programmable logic device. The Examiner in rejecting various depending claims acknowledges that the cited art does not teach the use of an SRAM PLD or an EEPROM PLD, but cites Michael Barr “How Programmable Logic Works” as describing the use of EEPROM and SRAM in programmable memory devices. The Applicants recognize that SRAM PLDs are well known, but use of SRAM PLDs in the context recited by the claims is believed to be novel and nonobvious.

Various dependent claims recite a PLD as the configurable device. Assuming that somehow the Examiner’s assertion that a flash memory or BIOS is a configurable device, it would make no sense at all combine Barr and Albrecht and use the PLD configurable device as a flash memory or BIOS. Both a flash memory and BIOS are meant to be inexpensive mechanisms for being persistent storage mechanisms for small amounts of data. Using a PLD as a flash memory or a BIOS would be entirely counterintuitive, and neither Barr nor Albrecht suggest such a use. A PLD includes programmable logic that can significantly slow processing. It would make no sense to use a PLD as a flash memory or BIOS.

CONCLUSION

In light of the above remarks relating to independent claims and certain dependent claims, the remaining dependent claims are believed allowable for at least the reasons noted above. Applicants believe that all pending claims are allowable. Should the Examiner believe that a telephone conference would expedite the prosecution of this application, please feel free to contact the undersigned at the telephone number set out below.

Respectfully submitted,
BEYER WEAVER LLP

/Godfrey Audrey Kwan/
Godfrey Audrey Kwan
Reg. No. 46,850

P.O. Box 70250
Oakland, CA 94612-0250
(510) 663-1100